LCA Cheatsheet

Setup - Runs in O(N*logN):

```
// fill parent[i] with the parent of node i. The parent of the root should be itself.
// fill depth[i] with the depth of node i

for (int i = 0; i<N; i++) {
	ancestor[i][0] = parent[i];
}

for (int k = 0; k < 17; k++) { // Replace 17 with log base2 of N
	for (int i = 0; i<N; i++) {
		ancestor[i][k+1] = ancestor[ancestor[i][k]][k];
	}
}
```

LCA Function - Runs in O(log N):

```
int LCA(int A, int B) {
	if (dep[A] > dep[B]) {
		swap(A, B); // in Java replace this with code to swap A and B
	}
	int d = dep[B] - dep[A];
	for (int i = 0; i < 17; i++) {
		if (d & (1<<i) ) {
			B = ancestor[B][i];
		}
	}
	if (A == B) {
		return A;
	}

	for (int i=16; i>= 0; i--) {
		if (ancestor[A][i] != ancestor[B][i]) {
			A = ancestor[A][i];
			B = ancestor[B][i];
		}
	}
	return parent[A];
}
```